Automatic Optimization of Domain Specific Languages for Weather and Climate Models

Tobias Wicky¹,², Carlos Osuna¹,², Oliver Fuhrer², Torsten Hoefler¹
¹ Department of Computer Science, ETH Zurich; ² Federal Office of Meteorology and Climatology MeteoSwiss, Switzerland

1 Introduction

- Speedup due to accelerators
- Developments speed suffers
- Maintenance effort increases
- Current DSLs are domain specific

Stella (currently operational)
- HPC knowledge still required
- Significant boilerplate code
- Focused on GPU as accelerator

2 Main Objectives

1. Creation of a toolchain for translating high level DSL code to optimized HPC code
2. Allow for architecture-independent development
3. Allow for development without exposing parallel concepts in the code
4. Allow for a descriptive, precise language with minimal boilerplate code
5. Optimization competitive to expert tuned code
6. Expose shared functionality to other high level DSLs via an Intermediate Representation

3 Frontend Code Examples

Code Examples

```c
# Example

void example_function(float x, float y) {
    float result = x + y;
    return result;
}
```

4 Results and discussion

Performance Results

Table 1. Lines of code for specific stencils

<table>
<thead>
<tr>
<th>Stencil</th>
<th>Lines of Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vert Advection</td>
<td>530</td>
</tr>
<tr>
<td>Fast Waves</td>
<td>4332</td>
</tr>
<tr>
<td>Diffusion</td>
<td>517</td>
</tr>
</tbody>
</table>

5 Conclusion

- functional toolchain
- significant reduction in code-size
- language independent of architecture
- visible performance gains
- intuitive and simple language for numerical methods

6 References

1. Michael Bader, Axel Bellet, Jochen Förstner, Detlev Majewski, Matthias Raschendorfer, and Thorsten Reinhardt.